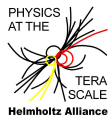


ILD Meeting Cracow 2013 – Latest News from WHIZARD

Jürgen R. Reuter

DESY Hamburg



ILD Workshop 2013, Cracow, Sep 26th, 2013

Apologies – and thanks for invitation



Przeprosiny – dziękuję za zaproszenie



The WHIZARD Event Generator – Release 2.1

- ▶ Multi-Channel Monte-Carlo integration
- ▶ Efficient phase space and event generation (weighted & unweighted)
- ▶ Optimized tree-level matrix elements (O'Mega)
 - $e^+e^- \rightarrow t\bar{t}H \rightarrow b\bar{b}b\bar{b}jj\ell\nu$ (110,000 diagrams)
 - $e^+e^- \rightarrow ZHH \rightarrow ZWWWW \rightarrow bb + 8j$ (12,000,000 diagrams)
 - $pp \rightarrow \ell\ell + nj, n = 0, 1, 2, 3, 4, \dots$ (2,100,000 diagrams with 4 jets + flavors)
 - $pp \rightarrow \tilde{\chi}_1^0 \tilde{\chi}_1^0 bbbb$ (32,000 diagrams, 22 color flows, $\sim 10,000$ PS channels)
 - $pp \rightarrow VVjj \rightarrow jj\ell\ell\nu\nu$ incl. anomalous TGC/QGC
 - Test case $gg \rightarrow 9g$ (224,000,000 diagrams)



WHIZARD 2.1.1 release: Sep. 18, 2012

Old series: WHIZARD 1.97 (development stopped with 1.94)

The WHIZARD team: F. Bach, [H. Boschmann], [F. Braam], B. Chokoufé, **W. Kilian**, **T. Ohl**, **JRR**, [S. Schmidt], [S. Schwertfeger], M. Sekulla, [C. Speckner], F. Staub, [M. Trudewind], C. Weiss, [D. Wiesler]

Web address: <http://projects.hepforge.org/whizard>

Standard Reference: [Kilian/Ohl/JRR, EPJC 71 \(2011\) 1742, arXiv:0708.4233](#)

The WHIZARD Event Generator – Release 2.2

- ▶ Multi-Channel Monte-Carlo integration
- ▶ Efficient phase space and event generation (weighted & unweighted)
- ▶ Optimized tree-level matrix elements (O'Mega)
 - $e^+e^- \rightarrow t\bar{t}H \rightarrow b\bar{b}b\bar{b}jj\ell\nu$ (110,000 diagrams)
 - $e^+e^- \rightarrow ZHH \rightarrow ZWWWW \rightarrow bb + 8j$ (12,000,000 diagrams)
 - $pp \rightarrow \ell\ell + nj, n = 0, 1, 2, 3, 4, \dots$ (2,100,000 diagrams with 4 jets + flavors)
 - $pp \rightarrow \tilde{\chi}_1^0 \tilde{\chi}_1^0 bbbb$ (32,000 diagrams, 22 color flows, $\sim 10,000$ PS channels)
 - $pp \rightarrow VVjj \rightarrow jj\ell\ell\nu\nu$ incl. anomalous TGC/QGC
 - Test case $gg \rightarrow 9g$ (224,000,000 diagrams)



WHIZARD 2.2.0_α-1 **release: Sep. 25, 2013**

Old series: WHIZARD 1.97 (development stopped with 1.94)

The WHIZARD team: F. Bach, [H. Boschmann], [F. Braam], B. Chokoufé, **W. Kilian**, **T. Ohl**, **JRR**, [S. Schmidt], [S. Schwertfeger], M. Sekulla, [C. Speckner], F. Staub, [M. Trudewind], C. Weiss, [D. Wiesler]

Web address: <http://projects.hepforge.org/whizard>

Standard Reference: [Kilian/Ohl/JRR, EPJC 71 \(2011\) 1742, arXiv:0708.4233](#)

The WHIZARD Event Generator – Release 2.2

- ▶ Multi-Channel Monte-Carlo integration
- ▶ Efficient phase space and event generation (weighted & unweighted)
- ▶ Optimized tree-level matrix elements (O'Mega)
 - $e^+e^- \rightarrow t\bar{t}H \rightarrow b\bar{b}b\bar{b}jj\ell\nu$ (110,000 diagrams)
 - $e^+e^- \rightarrow ZHH \rightarrow ZWWWW \rightarrow bb + 8j$ (12,000,000 diagrams)
 - $pp \rightarrow \ell\ell + nj, n = 0, 1, 2, 3, 4, \dots$ (2,100,000 diagrams with 4 jets + flavors)
 - $pp \rightarrow \tilde{\chi}_1^0 \tilde{\chi}_1^0 bbbb$ (32,000 diagrams, 22 color flows, $\sim 10,000$ PS channels)
 - $pp \rightarrow VVjj \rightarrow jj\ell\ell\nu\nu$ incl. anomalous TGC/QGC
 - Test case $gg \rightarrow 9g$ (224,000,000 diagrams)



WHIZARD 2.2.0

release: Nov. 11, 2013 (LCWS)

Old series: WHIZARD 1.97 (development stopped with 1.94)

The WHIZARD team: F. Bach, [H. Boschmann], [F. Braam], B. Chokoufé, **W. Kilian**, **T. Ohl**, **JRR**, [S. Schmidt], [S. Schwertfeger], M. Sekulla, [C. Speckner], F. Staub, [M. Trudewind], C. Weiss, [D. Wiesler]

Web address: <http://projects.hepforge.org/whizard>

Standard Reference: [Kilian/Ohl/JRR, EPJC 71 \(2011\) 1742, arXiv:0708.4233](#)

The WHIZARD Event Generator – Release 2.2

- ▶ Multi-Channel Monte-Carlo integration
- ▶ Efficient phase space and event generation (weighted & unweighted)
- ▶ Optimized tree-level matrix elements (O'Mega)

- $e^+e^- \rightarrow t\bar{t}H \rightarrow b\bar{b}b\bar{b}jj\ell\nu$ (110,000 diagrams)
- $e^+e^- \rightarrow ZHH \rightarrow ZWWWW \rightarrow bb + 8j$ (12,000,000 diagrams)
- $pp \rightarrow \ell\ell + nj, n = 0, 1, 2, 3, 4, \dots$ (2,100,000 diagrams with 4 jets + flavors)
- $pp \rightarrow \tilde{\chi}_1^0 \tilde{\chi}_1^0 bbbb$ (32,000 diagrams, 22 color flows, $\sim 10,000$ PS channels)
- $pp \rightarrow VVjj \rightarrow jj\ell\ell\nu\nu$ incl. anomalous TGC/QGC
- Test case $gg \rightarrow 9g$ (224,000,000 diagrams)



WHIZARD 2.2.0 release: Nov. 11, 2013 (LCWS)

Old series: WHIZARD 1.97 (development stopped with 1.94)

The WHIZARD team: F. Bach, [H. Boschmann], [F. Braam], B. Chokoufé, W. Kilian, T. Ohl, JRR, [S. Schmidt], [S. Schwertfeger], M. Sekulla, [C. Speckner], F. Staub, [M. Trudewind], C. Weiss, [D. Wiesler] + 3 bachelors

Web address: <http://projects.hepforge.org/whizard>

Standard Reference: Kilian/Ohl/JRR, EPJC 71 (2011) 1742, arXiv:0708.4233

WHIZARD 2: Status 2011/12 – Technical Features

- WHIZARD 2: code basically rewritten, only `Fortran 2003` and `O'Cam1`
- Clean modularization of code/(First) object-oriented implementation
- OpenMP parallelization
- Operation modes:
 - ▶ Dynamic linking (default mode) with on-the-fly generation of process code
 - ▶ Static linking (for batch clusters)
 - ▶ Library mode, callable from C/C++/Python/...
 - ▶ Interactive mode: WHIZARD works as a Shell – WHISH
- Standard conformance: uses `autotools: automake/autoconf/libtool`
- Large self test suite
- Version control (`svn`) at `HepForge`: use of ticket system and bug tracker
- Continuous integration system (`jenkins`) linked with `svn` repository

WHIZARD 2 – Installation and Run

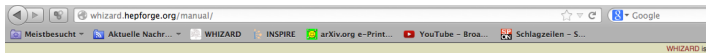
- ▶ Download WHIZARD from <http://www.hepforge.org/archive/whizard/whizard-2.1.1.tar.gz> and unpack it
- ▶ WHIZARD intended to be centrally installed on a system, e.g. in `/usr/local` (or locally on user account)
- ▶ Create build directory and configure
External programs (LHAPDF, StdHEP, HepMC) might need flags
- ▶ `make, make install`
- ▶ Create SINDARIN steering file (in any working directory)
- ▶ Run `whizard` (in working directory)
- ▶ **Supported event formats:** HepMC, StdHEP, LHEF, LHA, div. ASCII formats

```
O'Mega self tests:
make check-TESTS
PASS: test_omega95
PASS: test_omega95_bispinors
PASS: test_qed_eemm
PASS: ects
PASS: ward
PASS: compare_split_function
PASS: compare_split_module
=====
All 7 tests passed
=====
```

```
WHIZARD self tests:
make check-am
make check-TESTS
PASS: empty.run
PASS: vars.run
PASS: md5.run
[.....]
XFAIL: errors.run
PASS: extpar.run
PASS: susyhit.run
PASS: libs.run
PASS: qedtest.run
PASS: helicity.run
PASS: smtest.run
PASS: defaultcuts.run
PASS: restrictions.run
PASS: decays.run
PASS: alphas.run
PASS: colors.run
PASS: cuts.run
PASS: lhapdf.run
PASS: ilc.run
PASS: mssmtest.run
PASS: models.run
PASS: stdhep.run
PASS: stdhep_up.run
=====
```

```
All 53 tests behaved as expected (1 e
=====
```

WHIZARD Manual



- Home
- Downloads
- Wiki
- News
- ChangeLog
- Subversion
- Browser
- Tracker
- Internal

WHIZARD 2.1 A generic Monte-Carlo integration and event generation package for multi-particle processes MANUAL¹

Wolfgang Kilian,² Thorsten Ohl,³ Jürgen Reuter,⁴ Christian Speckner⁵

- Contents
- Introduction
 - Disclaimer
 - Overview
 - About examples in this manual
- Installation
 - Package Structure
 - Prerequisites
 - Installation
 - Working With WHIZARD
- Getting Started
 - Hello World
 - A Simple Calculation
- SINDARIN: Overview
 - The command language for WHIZARD
 - SINDARIN scripts
 - Errors
 - Statements
 - Control Structures
 - Expressions
 - Variables

Physics aspects/improvements in WHIZARD 2

- **SINDARIN** (Scripting **I**ntegration, **D**ata **A**nalysis, **R**esults display and **I**nterfaces) allows for arbitrary expressions for cuts and scales etc. (examples later)

```
cuts = any 5 degree < Theta < 175 degree  
      [select if abs (Eta) < eta_cut [lepton]]  
cuts = any E > 2 * mW [extract index 2  
                      [sort by Pt [lepton]]]
```

- New syntax for decays and chains:

```
process higgsstr = e1, E1 => (Z => e2, E2), (H => b, bbar)  
process wtf      = e1, E1 => (Z, h) + (Z, H) + (A, H)
```

- Process libraries: processes of different BSM models can be used in parallel
- Decay cascades including full spin correlations (cf. later)

- FeynRules interface

Christensen/Duhr/Fuks/JRR/Speckner, EPJC 72 (2012) 1990

- MLM jet matching

- Event-dependent scales in PDFs and running α_s

- Parton Shower: p_T -ordered and analytic

Kilian/JRR/Schmidt/Wiesler, JHEP 1204 (2012) 013

Structured Beams

► Lepton Colliders structured beams

- QED ISR (Skrzypek/Jadach, Kuraev/Fadin , incl. p_T distributions)
- arbitrarily polarized beams (density matrices)
- Beamstrahlung (CIRCE module) **more later**
- Photon collider spectra (CIRCE2 module)
- external beam spectra can be read in (files/generating code)
- QED FSR (e.g. YFS) not (yet) implemented (charged mesons/hadrons)

► Hadronic events/hadronic decays

- through PYTHIA interface [or HERWIG]

► Hadron Colliders structured beams

- LHAPDF interface
- Most prominent PDFs directly included
- QCD ISR and FSR (two different own implementations, interface to PYTHIA)
- Matching matrix elements/showers (MLM)
- Underlying event/multiple interactions

O'Mega: Optimal matrix elements

Ohl/JRR, 2001

 Ω

- [$\cdot$] Replace forest of tree diagrams by
Directed Acyclical Graph (DAG) of the algebraic expression (including color).

$$ab(ab + c) =$$

$$=$$

O'Mega: Optimal matrix elements

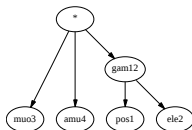
Ohl/JRR, 2001



- [$\cdot$] Replace forest of tree diagrams by
Directed Acyclical Graph (DAG) of the algebraic expression (including color).

$$ab(ab + c) =$$

- simplest examples: $e^+e^- \rightarrow \mu^+\mu^-$, and



O'Mega: Optimal matrix elements

Ohl/JRR, 2001

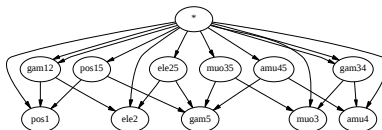
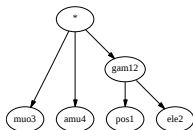


- [$\cdot$] Replace forest of tree diagrams by
Directed Acyclical Graph (DAG) of the algebraic expression (including color).

$$ab(ab + c) =$$

$$=$$

- simplest examples: $e^+e^- \rightarrow \mu^+\mu^-$, $e^+e^- \rightarrow \mu^+\mu^-\gamma$ and



O'Mega: Optimal matrix elements

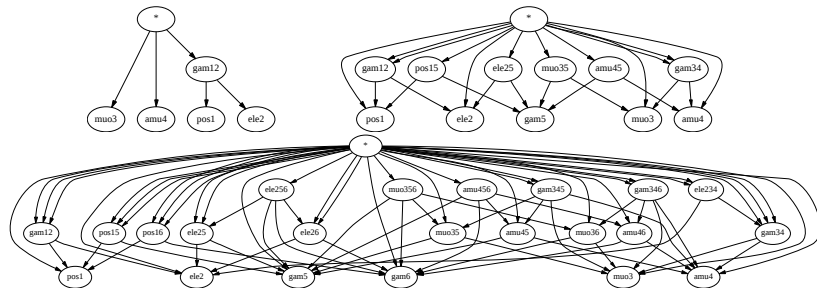
OHI/JRR, 2001



- [$\cdot$] Replace forest of tree diagrams by
Directed Acyclical Graph (DAG) of the algebraic expression (including color).

$$ab(ab + c) = \begin{array}{c} \times \\ \swarrow \quad \searrow \\ \begin{array}{cc} \times & + \\ \swarrow \quad \searrow & \swarrow \quad \searrow \\ a \quad b & \begin{array}{cc} \times & + \\ \swarrow \quad \searrow & \swarrow \quad \searrow \\ a \quad b & c \end{array} \end{array} = \begin{array}{c} \times \\ \swarrow \quad \searrow \\ \begin{array}{cc} \times & + \\ \swarrow \quad \searrow & \swarrow \quad \searrow \\ a \quad b & c \end{array} \end{array}$$

- simplest examples: $e^+e^- \rightarrow \mu^+\mu^-$, $e^+e^- \rightarrow \mu^+\mu^-\gamma$ and $e^+e^- \rightarrow \mu^+\mu^-\gamma\gamma$



Hard matrix elements: particle types

Possible particle types

- ▶ Spin 0 particles
- ▶ Spin 1/2 fermions (Majorana and Dirac)
Fermi statistics for both fermion-number conserving and violating cases
- ▶ Spin 1 particles
 - ▶ massive and massless
 - ▶ Unitarity and Feynman gauge
 - ▶ arbitrary R_ξ gauges
- ▶ Spin 3/2 particles (Majorana only, gravitinos)
- ▶ Spin 2 particles (massless and massive, gravitons)
- ▶ Dynamic particles vs. pure insertions
- ▶ Unphysical particles for Ward- and Slavnov-Taylor identities

Hard matrix elements: Lorentz structures

Hard-coded set of Lorentz structures

- Purely scalar couplings:

$$\phi^3, \quad \phi^4$$

- Scalar couplings to vectors:

$$gV^\mu \phi_1 \overleftrightarrow{\partial}_\mu \phi_2, \quad \phi V^2, \quad \phi^2 V^2, \quad \frac{1}{2} \phi F_{1,\mu\nu} F_2^{\mu\nu}, \quad \frac{1}{2} \phi F_{1,\mu\nu} \tilde{F}_2^{\mu\nu}, \quad \phi (i\partial_\mu V_1^\nu) (i\partial_\nu V_2^\mu)$$

- Pure vector couplings:

$$F_{\mu\nu} F^{\mu\nu}, \quad V_1^\mu ((i\partial_\nu V_2^\rho) \overleftrightarrow{\partial}_\mu (i\partial_\rho V_3^\nu)), \quad g F_1^{\mu\nu} F_{2,\nu\rho} F_{3,\mu}^\rho, \\ g/2 \cdot \epsilon^{\mu\nu\lambda\tau} F_{1,\mu\nu} F_{2,\tau\rho} F_{3,\lambda}^\rho$$

- Fermionic couplings to scalars:

$$g_S \bar{\psi}_1 S \psi_2, \quad g_P \bar{\psi}_1 P \gamma_5 \psi_2, \quad \bar{\psi}_1 \phi (g_S + g_P \gamma_5) \psi_2, \quad g_L \bar{\psi}_1 \phi (1 - \gamma_5) \psi_2, \\ g_R \bar{\psi}_1 \phi (1 + \gamma_5) \psi_2, \quad g_L \bar{\psi}_1 \phi (1 - \gamma_5) \psi_2 + g_R \bar{\psi}_1 \phi (1 + \gamma_5) \psi_2$$

- Fermionic couplings to vectors:

$$g_V \bar{\psi}_1 V \psi_2, \quad g_A \bar{\psi}_1 \gamma_5 V \psi_2, \quad \bar{\psi}_1 V (g_V - g_A \gamma_5) \psi_2, \quad g_L \bar{\psi}_1 V (1 - \gamma_5) \psi_2, \\ g_R \bar{\psi}_1 V (1 + \gamma_5) \psi_2, \quad g_L \bar{\psi}_1 V (1 - \gamma_5) \psi_2 + g_R \bar{\psi}_1 V (1 + \gamma_5) \psi_2$$

- ▶ Fermionic couplings in SUSY Ward identities (not listed here)
- ▶ Fermionic couplings to tensors:

$$g_T T_{\mu\nu} \bar{\psi}_1 [\gamma^\mu, \gamma^\nu] \psi_2$$

- ▶ Tensor couplings to vectors:

$$T^{\mu\nu} (V_{1,\mu} V_{2,\nu} + V_{1,\nu} V_{2,\mu}), \quad T^{\alpha\beta} (V_1^\mu i \overleftrightarrow{\partial}_\alpha i \overleftrightarrow{\partial}_\beta V_{2,\mu}, \\ T^{\alpha\beta} (V_1^\mu i \overleftrightarrow{\partial}_\beta (i \partial_\mu V_{2,\alpha}) + V_1^\mu i \overleftrightarrow{\partial}_\alpha (i \partial_\mu V_{2,\beta})) , \quad T^{\alpha\beta} ((i \partial^\mu V_1^\nu) i \overleftrightarrow{\partial}_\alpha i \overleftrightarrow{\partial}_\beta (i \partial_\nu V_{2,\mu}))$$

- ▶ Gravitino couplings:

$$\bar{\psi} \gamma^\mu S \psi_\mu, \quad \bar{\psi} \gamma^\mu \not{k}_S S \psi_\mu, \quad \bar{\psi} \gamma^\mu \gamma^5 P \not{k}_P \psi_\mu, \quad \bar{\psi} \gamma^5 \gamma^\mu [\not{k}_V, \not{V}] \psi_\mu \text{ etc.}$$

and many more to fill your advent calendar.....

- ▶ Completely general Lorentz structures:
work in progress, to appear in version 2.2

Hard matrix elements: Color structures

Possible Color structures

- ▶ All $SU(N)$ gauge theories supported, but specialize to $N = 3$
- ▶ Color flow formalism
Kilian/Ohl/JRR/Speckner, 2011
Stelzer/Willenbrock, 2003;
- ▶ Fundamental representations: $\mathbf{3}, \bar{\mathbf{3}}$
- ▶ Adjoint representation: $\mathbf{8}$
- ▶ Covers all interactions e.g. in SUSY and extra dimensions
- ▶ **in preparation:** generalized color structures with reps. $\mathbf{6}, \bar{\mathbf{6}}, \mathbf{10}, \bar{\mathbf{10}}$
as well as $\epsilon_{ijk}\phi_i\phi_j\phi_k$ couplings **to appear in version 2.2.x**

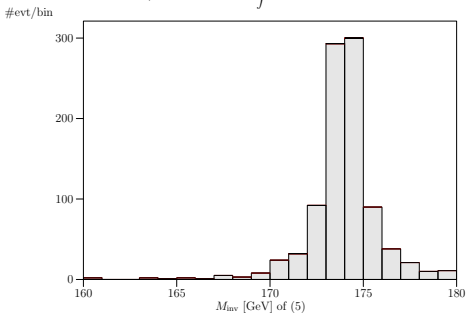
WHIZARD histograms

WHIZARD data analysis

March 16, 2007

Process: qqttdec ($u\bar{u} \rightarrow b\bar{b}W^+W^-$)

$$\sqrt{s} = 500.0 \text{ GeV} \quad \int \mathcal{L} = 0.2754 \times 10^{-01} \text{ fb}^{-1}$$



$\sigma_{\text{tot}} = 36305. \pm 310. \text{ fb} \quad [\pm 0.85 \%]$
 $n_{\text{evt, tot}} = 1000$
 $\sigma_{\text{cut}} = 36305. \pm 0.115 \times 10^{+04} \text{ fb} \quad [\pm 3.16 \%]$
 $n_{\text{evt, cut}} = 1000 \quad [100.00 \%]$

New completely general syntax in WHIZARD 2.x

```

$title = "Jet Energy in $pp\to \ell\ell\bar{\nu}\nu j$"
$x_label = "$E$/GeV"
histogram e_jet (0 GeV, 80 GeV, 2 GeV)
analysis = record pt_lepton (eval Pt [extract index 1 [sort by Pt [lepton]]]);
           record pt_jet (eval Pt [extract index 1 [sort by Pt [jet]]]);
           record e_lepton (eval E [extract index 1 [sort by Pt [lepton]]]);
           record e_jet (eval E [extract index 1 [sort by Pt [jet]]])

```

WHIZARD – Overview over BSM Models

MODEL TYPE	with CKM matrix	trivial CKM
QED with e, μ, τ, γ	—	QED
QCD with d, u, s, c, b, t, g	—	QCD
Standard Model	SM_CKM	SM
SM with anomalous gauge coupl.	SM_ac_CKM	SM_ac
SM with anomalous top coupl.	SMtop_CKM	SMtop
SM with K matrix	—	SM_KM
MSSM	MSSM_CKM	MSSM
MSSM with gravitinos	—	MSSM_Grav
NMSSM	NMSSM_CKM	NMSSM
extended SUSY models	—	PS/E/SSM
Littlest Higgs	—	Littlest
Littlest Higgs with ungauged $U(1)$	—	Littlest_Eta
Littlest Higgs with T parity	—	Littlest_Tpar
Simplest Little Higgs (anomaly-free)	—	Simplest
Simplest Little Higgs (universal)	—	Simplest_univ
3-site model	—	Thresh1
UED	—	UED
SM with Z'	—	Zprime
SM with gravitino and photino	—	GravTest
Augmentable SM template	—	Template

new models easily: FeynRules interface [Christensen/Duhr/Fuks/JRR/Speckner, 1010.3251](#)

Interface to SARAH in the SUSY Toolbox [Staub, 0909.2863; Ohl/Porod/Speckner/Staub, 1109.5147](#)

Input files: Basic features

```
model = SM
```

```
process halloween = E1, e1 => t, tbar, H
```

```
compile
```

```
sqrts = 500
```

```
beams = E1, e1 => circel => isr
```

```
integrate (susybg) { iterations = 5:10000, 2:10000 }
```

```
n_events = 10000
```

```
simulate (full) {  
}
```

Example: LHC SUSY cascade decays, Input File

```
model = MSSM

process dec_su_q = su1 => u, neu2
process dec_neu_sl2 = neu2 => SE12, e1

process susybg = u,U => SU1, su1
process full = u, U => SU1, u, e1, SE12

compile

?slha_read_decays = true
read_slha("spslap_decays.slha")

integrate (dec_su_q, dec_neu_sl2) { iterations = 1:1000 }

sqrts = 14000
beams = p, p => lhpdf

integrate (susybg) { iterations = 5:10000, 2:10000 }
integrate (full)

n_events = 10000

$title = "Full process"
$description =
  "$p + p \to u + \bar{u} \to \bar{u} + u + \tilde{e}_{12}^+ + e^- $"
$xlabel = "$M_{\rm inv}(ue^-) $"
histogram inv_mass1_full (0,600,20)

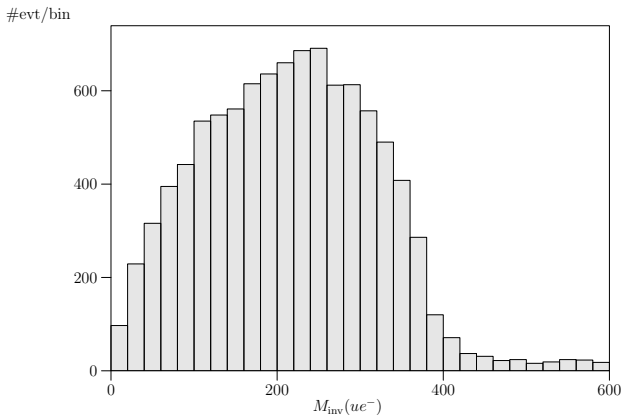
simulate (full) {
  $sample = "casc_dec_full"
  analysis =
    record inv_mass1_full (eval M / 1 GeV [combine[u,e1]])
}

compile_analysis
$analysis_filename = "casc_dec"
write_analysis
```

Example: LHC SUSY cascade decays

$$p + p \rightarrow \tilde{u}^* + \tilde{u} \rightarrow \tilde{u}^* + u + \tilde{e}^+ + e^-$$

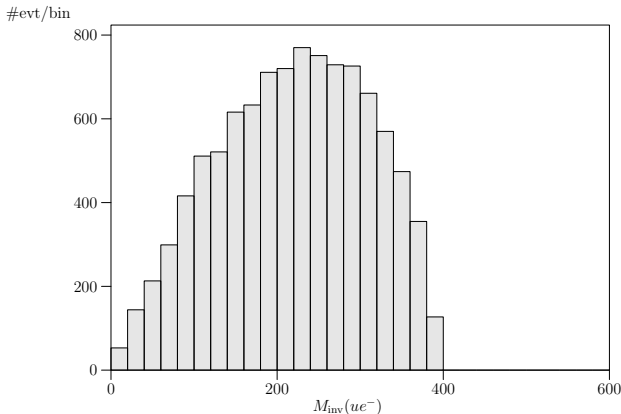
► Full process:



Example: LHC SUSY cascade decays

$$p + p \rightarrow \tilde{u}^* + \tilde{u} \rightarrow \tilde{u}^* + u + \tilde{e}^+ + e^-$$

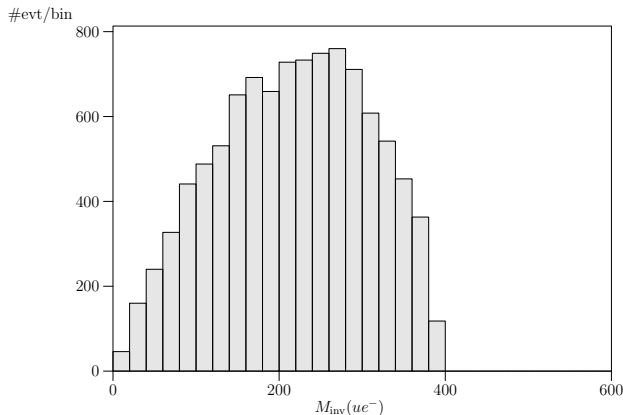
► **Factorized process w/ full spin correlations:**



Example: LHC SUSY cascade decays

$$p + p \rightarrow \tilde{u}^* + \tilde{u} \rightarrow \tilde{u}^* + u + \tilde{e}^+ + e^-$$

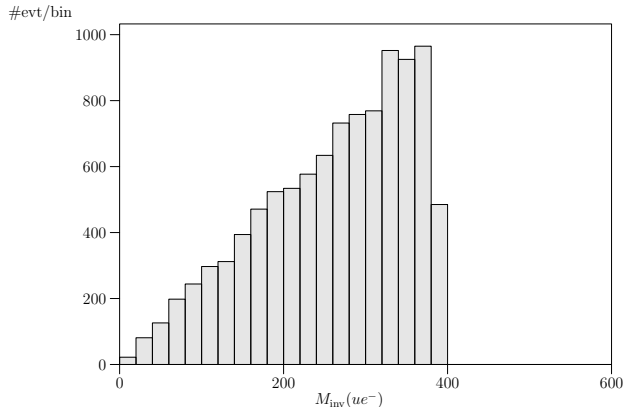
► **Factorized process w/ classical spin correlations:**



Example: LHC SUSY cascade decays

$$p + p \rightarrow \tilde{u}^* + \tilde{u} \rightarrow \tilde{u}^* + u + \tilde{e}^+ + e^-$$

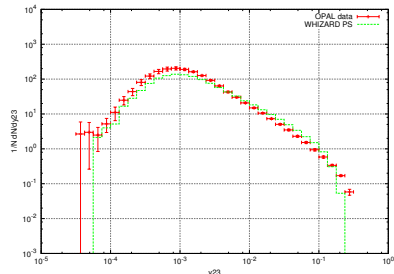
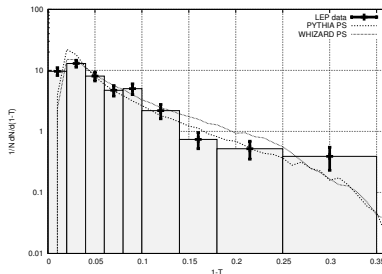
► **Factorized process w/ no spin correlations:**



Analytic Parton Shower

JRR/Schmidt/Wiesler, JHEP 2012

- ▶ **Analytic Parton Shower:**
 - no shower veto: shower history is exactly known
 - allows reweighting and maybe more reliable error estimate
- ▶ new algorithm for initial state QCD radiation

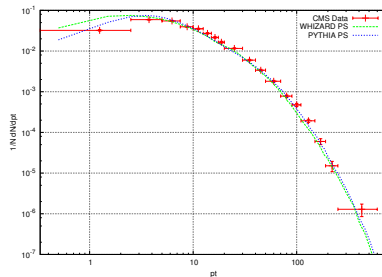
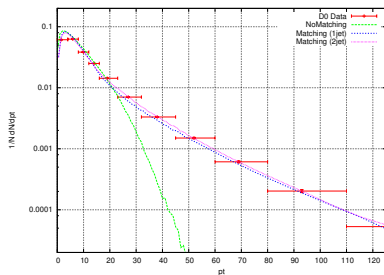


- ▶ matching with hard matrix elements, no "power-shower"

Analytic Parton Shower

JRR/Schmidt/Wiesler, JHEP 2012

- ▶ **Analytic Parton Shower:**
 - no shower veto: shower history is exactly known
 - allows reweighting and maybe more reliable error estimate
- ▶ new algorithm for initial state QCD radiation



- ▶ matching with hard matrix elements, no "power-shower"

Status of NLO development in WHIZARD

► BLHA interface: workflow

Speckner, 2012

1. Process definition in SINDARIN $\Rightarrow$ WHIZARD writes contract file
2. NLO generator generates code, WHIZARD reads contract
3. NLO matrix element loaded as shared library

► First implementation: interfacing GoSAM and FeynArts

► Automatic generation of dipole subtraction terms

Speckner, 2012; JRR/Weiss, 2013/14

- proof-of-concept code in WHIZARD 2.1
- implementation in the context of the revised WHIZARD 2.2 core

First example: $u\bar{u} \rightarrow \mu^-\bar{\nu}_\mu e^+\nu_e$

Input:

```
real mreg = 1 GeV

process test = u, ubar => "mu-", numubar, "e+", nue {
  $method = "dipole_integrated_qed"
  soft_mass_regulator = mreg
  collinear_mass_regulators = mreg, mreg, mreg, 0, mreg, 0
}

me = 0
mmu = 0
alpha_qed = 1. / alpha_em_i

sqrts = 500 GeV

integrate (test) {iterations = 5:10000, 5:20000}
```

Result:

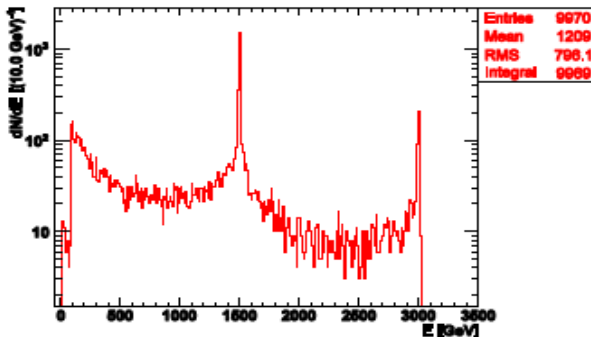
```
| Integrating process 'test':
|=====|
| It      Calls  Integral[fb]  Error[fb]  Err[%]   Acc  Eff[%]  Chi2 N[It] |
|=====|
| 10      100000  1.9794090E+00  3.16E-03  0.16    0.50  12.33   0.12  5   |
|=====|
```

Simulating Linear Colliders

- ▶ High-Energy Linear Lepton Collider (250/350/500/1000/2000/3000 GeV)
- ▶ **ISR, beamstrahlung, strong fields** (CLIC)
- ▶ Exhaustive support for these effects in WHIZARD (close collaboration with all LC groups)
- ▶ Prime Example $e^+e^- \rightarrow b\bar{b}$:

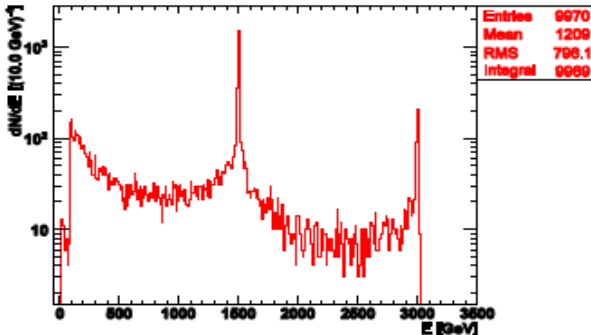
Simulating Linear Colliders

- ▶ High-Energy Linear Lepton Collider (250/350/500/1000/2000/3000 GeV)
- ▶ **ISR, beamstrahlung, strong fields** (CLIC)
- ▶ Exhaustive support for these effects in WHIZARD (close collaboration with all LC groups)
- ▶ Prime Example $e^+e^- \rightarrow b\bar{b}$:



Simulating Linear Colliders

- ▶ High-Energy Linear Lepton Collider (250/350/500/1000/2000/3000 GeV)
- ▶ **ISR, beamstrahlung, strong fields** (CLIC)
- ▶ Exhaustive support for these effects in WHIZARD (close collaboration with all LC groups)
- ▶ Prime Example $e^+e^- \rightarrow b\bar{b}$:



Luminosity spectrum picks up the Z resonance!

News 2013/early 2014: upcoming official release 2.2.x

- Lost of WHIZARD members 2012: some features delayed in 2013
(pre-)release version **2.2.0- α -1** available!

News 2013/early 2014: upcoming official release 2.2.x

- Lost of WHIZARD members 2012: some features delayed in 2013
(pre-)release version **2.2.0- α -1** available!
- WHIZARD core: insert an extra abstraction layer, consistently separate interface from implementation **Complete object orientation**
 - ▶ **Replaceable modules** with well-defined interface: matrix-elements, beam structure, phase space, integration, decays, shower, ...
 - ▶ Much easier to contribute new parts to the code $\Rightarrow$ **Industrialization**
 - ▶ Framework for testing ideas and algorithms
 - ▶ Technical changes hidden from the user
 - ▶ Modern programming techniques: unit tests, pair programming etc.
 - ▶ Much better self checks, regression testing and maintainability

News 2013/early 2014: upcoming official release 2.2.x

- Lost of WHIZARD members 2012: some features delayed in 2013
(pre-)release version **2.2.0- α -1** available!
- WHIZARD core: insert an extra abstraction layer, consistently separate interface from implementation **Complete object orientation**
 - ▶ **Replaceable modules** with well-defined interface: matrix-elements, beam structure, phase space, integration, decays, shower, ...
 - ▶ Much easier to contribute new parts to the code $\Rightarrow$ **Industrialization**
 - ▶ Framework for testing ideas and algorithms
 - ▶ Technical changes hidden from the user
 - ▶ Modern programming techniques: unit tests, pair programming etc.
 - ▶ Much better self checks, regression testing and maintainability
- Revised model for BSM interactions of **electroweak vector bosons** (w/ light Higgs)
- Process containers: inclusive production samples (e.g. SUSY)
- Automatic generation of **decays**, depending on the model

News 2013/early 2014: upcoming official release 2.2.x

- Lost of WHIZARD members 2012: some features delayed in 2013
(pre-)release version **2.2.0- α -1** available!
- WHIZARD core: insert an extra abstraction layer, consistently separate interface from implementation **Complete object orientation**
 - ▶ **Replaceable modules** with well-defined interface: matrix-elements, beam structure, phase space, integration, decays, shower, ...
 - ▶ Much easier to contribute new parts to the code $\Rightarrow$ **Industrialization**
 - ▶ Framework for testing ideas and algorithms
 - ▶ Technical changes hidden from the user
 - ▶ Modern programming techniques: unit tests, pair programming etc.
 - ▶ Much better self checks, regression testing and maintainability
- Revised model for BSM interactions of **electroweak vector bosons** (w/ light Higgs)
- Process containers: inclusive production samples (e.g. SUSY)
- Automatic generation of **decays**, depending on the model
- Easy cascades: "e+", "e-" $\Rightarrow$ (t $\Rightarrow$ ("W+" $\Rightarrow$ "mu+", numu), b), tbar
- Specification of QCD and electroweak order
- Improvements to the **SINDARIN** steering language

New (LC-related) features / Plans

- **Lumi-linker** interface courtesy of T. Barklow
- **LCIO support** (C++ interface) courtesy of F. Gaede
- Support for **ILC beam spectra within CIRCE1** courtesy of G. Wilson
- Module for **CLIC beam spectra** provided by S. Poss

New (LC-related) features / Plans

- **Lumi-linker** interface courtesy of T. Barklow
- **LCIO support** (C++ interface) courtesy of F. Gaede
- Support for **ILC beam spectra within CIRCE1** courtesy of G. Wilson
- Module for **CLIC beam spectra** provided by S. Poss
- BSM: **general Lorentz structures** in matrix-element generator (O'Mega)
- Parsing of UFO files

New (LC-related) features / Plans

- **Lumi-linker** interface courtesy of T. Barklow
- **LCIO support** (C++ interface) courtesy of F. Gaede
- Support for **ILC beam spectra within CIRCE1** courtesy of G. Wilson
- Module for **CLIC beam spectra** provided by S. Poss
- BSM: **general Lorentz structures** in matrix-element generator (O'Mega)
- Parsing of UFO files
- Working on performance gain: multi-leg, parallelization, smaller expressions etc.

New (LC-related) features / Plans

- **Lumi-linker** interface courtesy of T. Barklow
- **LCIO support** (C++ interface) courtesy of F. Gaede
- Support for **ILC beam spectra within CIRCE1** courtesy of G. Wilson
- Module for **CLIC beam spectra** provided by S. Poss
- BSM: **general Lorentz structures** in matrix-element generator (O'Mega)
- Parsing of UFO files
- Working on performance gain: multi-leg, parallelization, smaller expressions etc.
- ▶ **CKKW(-L) matching for jets** Prestel/JRR/Weiss, ca. 2014
- ▶ **Matched $e^+e^- \rightarrow$ jets at LO and NLO, POWHEG box formalism** Prestel/JRR/Weiss, ca. 2014

New (LC-related) features / Plans

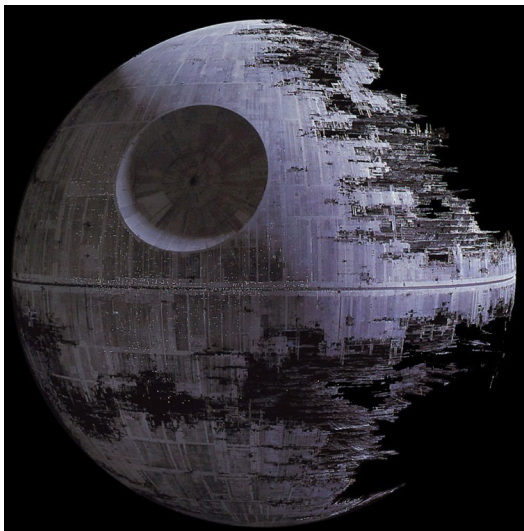
- **Lumi-linker** interface courtesy of T. Barklow
- **LCIO support** (C++ interface) courtesy of F. Gaede
- Support for **ILC beam spectra within CIRCE1** courtesy of G. Wilson
- Module for **CLIC beam spectra** provided by S. Poss
- BSM: **general Lorentz structures** in matrix-element generator (O'Mega)
- Parsing of UFO files
- Working on performance gain: multi-leg, parallelization, smaller expressions etc.
- ▶ **CKKW(-L) matching for jets** Prestel/JRR/Weiss, ca. 2014
- ▶ **Matched $e^+e^- \rightarrow$ jets at LO and NLO, POWHEG box formalism** Prestel/JRR/Weiss, ca. 2014
- ▶ **Threshold resummation for $e^+e^- \rightarrow W^+W^-, t\bar{t}$ etc.** Bach/JRR, ca. 2014

Status of refactoring:

Well, what shall I say ...

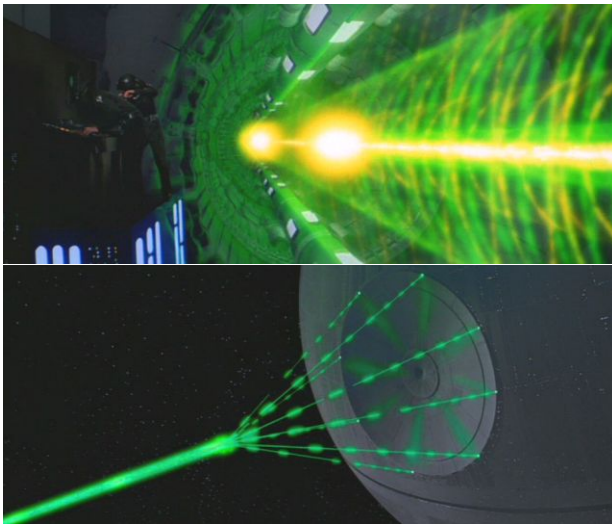
Status of refactoring:

Well, what shall I say ...



Status of refactoring:

ok, LC features have pretty high priority



Summary and Outlook

- ▶ **WHIZARD 2** for ILC (and LHC) physics
- ▶ **Versatile, user-friendly tool for SM & BSM physics**
- ▶ **Highest-possible support for LC beam structures**
- ▶ Covers the whole SM, and most possible paths beyond
- ▶ Steered via the HepForge page:
<http://projects.hepforge.org/whizard>
- ▶ Expect continuous improvement



Summary and Outlook

- ▶ **WHIZARD 2** for ILC (and LHC) physics
- ▶ **Versatile, user-friendly tool for SM & BSM physics**
- ▶ **Highest-possible support for LC beam structures**
- ▶ Covers the whole SM, and most possible paths beyond
- ▶ Steered via the HepForge page:
<http://projects.hepforge.org/whizard>
- ▶ Expect continuous improvement
 - WHIZARD 2.2.0- α -1 available now



Summary and Outlook

- ▶ **WHIZARD 2** for ILC (and LHC) physics
- ▶ **Versatile, user-friendly tool for SM & BSM physics**
- ▶ **Highest-possible support for LC beam structures**
- ▶ Covers the whole SM, and most possible paths beyond
- ▶ Steered via the HepForge page:
<http://projects.hepforge.org/whizard>
- ▶ Expect continuous improvement
 - WHIZARD 2.2.0_α-1 available now
 - WHIZARD 2.2.0_α-2 for Dresden aQGC workshop



Summary and Outlook

- ▶ **WHIZARD 2** for ILC (and LHC) physics
- ▶ **Versatile, user-friendly tool for SM & BSM physics**
- ▶ **Highest-possible support for LC beam structures**
- ▶ Covers the whole SM, and most possible paths beyond
- ▶ Steered via the HepForge page:
<http://projects.hepforge.org/whizard>
- ▶ Expect continuous improvement
 - WHIZARD 2.2.0_α-1 available now
 - WHIZARD 2.2.0_α-2 for Dresden aQGC workshop
 - WHIZARD 2.2.0_β for SiD workshop



Summary and Outlook

- ▶ **WHIZARD 2** for ILC (and LHC) physics
- ▶ **Versatile, user-friendly tool for SM & BSM physics**
- ▶ **Highest-possible support for LC beam structures**
- ▶ Covers the whole SM, and most possible paths beyond
- ▶ Steered via the HepForge page:
<http://projects.hepforge.org/whizard>
- ▶ Expect continuous improvement
 - WHIZARD 2.2.0_α-1 available now
 - WHIZARD 2.2.0_α-2 for Dresden aQGC workshop
 - WHIZARD 2.2.0_β for SiD workshop
 - WHIZARD 2.2.0 for LCWS 2013, Tokio



as usual: **we're open to users wish list!**

whizard@desy.de

We do our best ...

We do our best ...

to make it fit for you and run on your system!

